

# 基于目标和特征的视频可搜索 加密系统设计与实现

作者姓名: 郑 宇  
指导教师: 付 冲 教授  
校外导师: 啊啊啊 教授  
单位名称: 计算机科学与工程学院  
专业名称: 通信工程

东 北 大 学  
2018 年 6 月

# **Design and Implementation of Searchable Cryptosystem for Video from Object Detection and Feature Extraction**

by ZHENG Yu

Supervisor: Professor FU Chong  
Co-supervisor: Sheeeee M.M. MMM (CUHK)

**Northeastern University**

**June 2018**

## 毕业设计（论文）任务书

毕业设计（论文）题目：

基于目标和特征的视频可搜索加密系统设计与实现

设计(论文)的基本内容：

为了实现搜索加密的视频，本论文拟设计开创性可通用的视频可搜索加密系统，拟通过关键字搜索、相似性搜索和相似性排序，可以根据用户所提交的图片/视频搜索到按相关性排序的多个视频，拟完成内容如下：

1. 学习密码学相关课程和书籍，看近期信息检索和可搜索加密的论文以了解背景；
2. 拟通过简单结合两个领域的内容首先实现一个简单的视频可搜索加密系统作为可行解，视频-帧-SIFT 特征-Hash-加密-比 Hamming 距离；
3. 拟进行改进优化，设计一个较为通用的视频可搜索加密系统作为较优解，预处理-类关键字搜索-特征相似搜索-特征相似排序；
4. 总结归纳，找出后续的思路和需要进行的工作。

毕业设计（论文）专题部分：

题目：\_\_\_\_\_

设计或论文专题的基本内容：

学生接受毕业设计（论文）题目日期

第 1 周

指导教师签字：

2018 年 3 月 5 日

## 致学弟学妹

学弟学妹们好，我是14级学姐郑宇，这是我第一次尝试搭建latex模板也是第一次使用Latex，非常拙劣简单，还有很多瑕疵！希望有能力的学弟学妹能供一起改进我们的模板，让我们的模板越来越好，祝母校越来越强大！我已经把模板挂在Github上，真心希望有能力的学弟学妹可以一起改进模板，网址为：<https://github.com/YuZhengCUHK/NortheasternUniversityLatexTemplate>，大家可以一起在这里讨论更新提问。

本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新

本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新

本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新

本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新本段留给学弟学妹一起更新

关键字：我最可爱；我最可爱；相似性搜索；特征向量.

## **Abstract**

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

**Keywords:** Video; Dynamic Search; Similarity Search; Feature Vector.

# 目录

毕业设计（论文）任务书.....	I
摘要.....	II
ABSTRACT.....	III
<b>第1章 引言</b> .....	1
1.1 研究的背景和来源.....	1
1.2 相关研究现状.....	2
1.2.1 啊啊啊啊研究现状.....	2
1.2.2 可搜索加密研究现状.....	3
1.3 本文主要贡献.....	5
1.4 论文结构.....	6
<b>第2章 密码学基础介绍</b> .....	8
2.1 流加密.....	8
2.2 块加密.....	9
2.3 其他加密技术.....	13
<b>第3章 系统模型</b> .....	20
3.1 符号定义.....	20
3.2 系统框架.....	21
3.3 系统功能及特性.....	22
<b>第4章 系统构建</b> .....	26
4.1 视频预处理过程.....	26
4.1.1 视频预处理框架.....	26
4.1.2 视频预处理核心算法.....	27
4.2 动态关键字搜索.....	31
4.2.1 啊啊啊啊啊搜索框架.....	31
4.2.2 啊啊啊啊啊搜索核心算法.....	32
4.3 相似搜索及啊啊.....	34
4.3.1 啊啊啊啊啊框架.....	34
4.3.2 啊啊啊啊及啊啊核心算法.....	36
<b>第5章 结果展示</b> .....	40
5.1 测试环境.....	40
5.2 测试结果.....	40

第6章 总结及展望.....	44
6.1 本文技术总结.....	44
6.2 个人心得及展望.....	45
第7章 参考文献.....	48
第8章 致谢.....	54
第9章 本科学业成果.....	56

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivations . . . . .	1
1.1.1	Searchable Encryption . . . . .	1
1.2	Contributions . . . . .	2
1.3	This Thesis . . . . .	3
<b>2</b>	<b>Cryptography</b>	<b>5</b>
2.1	Stream Cipher . . . . .	5
2.2	Block Cipher . . . . .	6
2.3	Other Ciphers . . . . .	8
<b>3</b>	<b>System Model</b>	<b>11</b>
3.1	Preliminaries and Notations . . . . .	11
3.2	Overview . . . . .	12
3.3	Functions and Characteristics . . . . .	12
<b>4</b>	<b>Construction</b>	<b>15</b>
4.1	Video Preprocess . . . . .	15
4.1.1	Framework . . . . .	15
4.1.2	Algorithms . . . . .	16
4.2	First-step Dynamic Keyword Search . . . . .	17
4.2.1	Framework . . . . .	17
4.2.2	Algorithms . . . . .	17
4.3	Second-step Similarity Search . . . . .	17
4.3.1	Framework . . . . .	17
4.3.2	Algorithms . . . . .	18
<b>5</b>	<b>Implementation</b>	<b>19</b>
5.1	Environment . . . . .	19
5.2	Results . . . . .	19
<b>6</b>	<b>Conclusions and Outlook</b>	<b>23</b>
6.1	Conclusions . . . . .	23
6.2	Reflection and Outlook . . . . .	23
	<b>References</b>	<b>25</b>
	<b>Acknowledgement</b>	<b>27</b>
	<b>Achievements</b>	<b>28</b>



# 1. Introduction

## 1.1. Motivations

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

### 1.1.1. Searchable Encryption

**Keyword Search** With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. In addition, some other improved multi-keyword schemes [1], [2] also have been proposed in very recent.

There are some other similarity search schemes such as [3], [4], [5] [6] from other communities which also investigate this problem.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

**Dynamic Search** With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet,

no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

***Forward and Backward Privacy*** With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

## 1.2. Contributions

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

### 1.3. This Thesis

*Section 1:* With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

*Section 2:* With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

*Section 3:* With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.



## 2. Cryptography

To understand this topic (or thesis) better, we need to learn some preliminary knowledges in cryptography. In this section, important knowledges from reference books are introduced one by one, which can also be seen as outlines of my self-learning. The main reference books are "A Graduate Course in Applied Cryptography" [7] and "Introduction to Modern Cryptography" [8].

### 2.1. Stream Cipher

**A Shannon Cipher** A Shannon cipher is a pair  $\mathcal{E} = (E, D)$  of functions, in which  $\mathcal{E}$  is defined over  $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ .  $\mathcal{K}$  is the set of all keys (the key space),  $\mathcal{M}$  is the set of all messages (the message space), and that  $\mathcal{C}$  is the set of all ciphertexts (the ciphertext space). With this notation, we can write:

$$E : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C} \quad (2.1)$$

$$D : \mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M} \quad (2.2)$$

**Perfect Security** Let  $\mathcal{E} = (E, D)$  be a Shannon cipher defined over  $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ . Consider a probabilistic experiment in which the random variable  $k'$  is uniformly distributed over  $\mathcal{K}$ . If for all  $m_0, m_1 \in \mathcal{M}$ , and all  $c \in \mathcal{C}$ , we have

$$\Pr[E(k', m_0) = c] = \Pr[E(k', m_1) = c] \quad (2.3)$$

then we say that  $\mathcal{E}$  is a perfectly secure Shannon cipher.

**Shannon's Theorem** Let  $\mathcal{E} = (E, D)$  be a Shannon cipher defined over  $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ . If  $\mathcal{E}$  is perfectly secure, then  $|\mathcal{K}| \geq |\mathcal{M}|$ .

**Semantic Security** A cipher  $\mathcal{E}$  is semantically secure if for all efficient adversaries  $\mathcal{A}$ , the value  $\text{SSadv}[\mathcal{A}, \mathcal{E}]$  is negligible, where  $\mathcal{A}$ 's semantic security with respect to  $\mathcal{E}$  is defined as

$$\text{SSadv}[\mathcal{A}, \mathcal{E}] := |\Pr[W_0] - \Pr[W_1]| \quad (2.4)$$

**Security Against Message Recovery** A cipher  $\mathcal{E}$  is secure against message recovery if for all efficient adversaries  $\mathcal{A}$ , the value  $\text{MRadv}[\mathcal{A}, \mathcal{E}]$  is negligible, where  $\mathcal{A}$ 's message recovery advantage with respect to  $\mathcal{E}$  is defined as

$$\text{MRadv}[\mathcal{A}, \mathcal{E}] := |\Pr[W] - 1/|\mathcal{M}|| \quad (2.5)$$

**Parity Prediction** A cipher  $\mathcal{E}$  is secure against parity prediction if for all efficient adversaries  $\mathcal{A}$ , the value  $\text{Parityadv}[\mathcal{A}, \mathcal{E}]$  is negligible, where  $\mathcal{A}$ 's parity

prediction advantage with respect to  $\mathcal{E}$  is defined as

$$\text{Parityadv}[\mathcal{A}, \mathcal{E}] := |\Pr[W] - 1/2| \quad (2.6)$$

**Pseudo-random Generator** A pseudo-random generator, or PRG for short, is an efficient, deterministic algorithm  $G$  that, given as input a seed  $s$ , computes an output  $r$ . The seeds comes from a finite seed space  $\mathcal{S} = \{0, 1\}^l$  and the output  $r$  belongs to a finite output space  $\mathcal{R} = \{0, 1\}^L$ . We say that  $G$  is a PRG defined over  $(\mathcal{S}, \mathcal{R})$ .

**Secure PRG** A PRG  $G$  is secure if the value  $\text{PRGadv}[\mathcal{A}, G]$  is negligible for all efficient adversaries  $\mathcal{A}$ , where  $\mathcal{A}$ 's advantage with respect to  $G$  is defined as

$$\text{PRGadv}[\mathcal{A}, G] := |\Pr[W_0] - \Pr[W_1]| \quad (2.7)$$

**Stream Ciphers** Let  $G$  be a PRG defined over  $(\{0, 1\}^l, \{0, 1\}^L)$ ; that is,  $G$  stretches an  $l$ -bit seed to an  $L$ -bit output. The stream cipher  $\mathcal{E} = (E, D)$  constructed from  $G$  is defined over  $(\{0, 1\}^l, \{0, 1\}^L)$ ; for  $s \in \{0, 1\}^l$  and  $m, c \in \{0, 1\}^L$ , encryption and decryption are defined as follows: if  $|m| = v$ , then

$$|E(s, m) := G(s)[0..v-1] \oplus m \quad (2.8)$$

and if  $|c| = v$ , then

$$D(s, c) := G(s)[0..v-1] \oplus c \quad (2.9)$$

**Stream Cipher Limitations** The two-time pad is insecure. In particular, a stream cipher key should never be used to encrypt more than one message.

**Computational Indistinguishability** Distributions  $P_0$  and  $P_1$  are called computational indistinguishability if the value  $\text{Distadv}[\mathcal{A}, P_0, P_1]$  is negligible for all efficient adversaries  $\mathcal{A}$ , where  $\mathcal{A}$ 's advantage with respect to  $P_0, P_1$  is defined as

$$\text{Distadv}[\mathcal{A}, P_0, P_1] := |\Pr[W_0] - \Pr[W_1]| \quad (2.10)$$

**Statistical Distance** Suppose  $P_0$  and  $P_1$  are probability distributions on a finite set  $\mathcal{R}$ . Then their statistical distance is defined as

$$\Delta[P_0, P_1] := \frac{1}{2} \sum_{r \in \mathcal{R}} |P_0(r) - P_1(r)|. \quad (2.11)$$

## 2.2. Block Cipher

**Block Cipher** Functionally, a block cipher is a deterministic cipher  $\mathcal{E} = (E, D)$  whose message space and ciphertext space are the same (finite) set  $\mathcal{X}$ . If the key space of  $\mathcal{E}$  is  $\mathcal{K}$ , we say that  $\mathcal{E}$  is a block cipher defined over  $(\mathcal{K}, \mathcal{X})$ . We call an element  $x \in \mathcal{X}$  a data block, and refer to  $\mathcal{X}$  as the data block space of  $\mathcal{E}$ .

For every fixed key  $k \in \mathcal{K}$ , we can define the function  $f_k := E(k, \cdot)$ ; that is,

$f_k : \mathcal{X} \rightarrow \mathcal{X}$  sends  $x \in \mathcal{X}$  to  $E(k, x) \in \mathcal{X}$ . The usual correctness requirement for any cipher implies that for every fixed key  $k$ , the function  $f_k$  is one-to-one, and as  $\mathcal{X}$  is finite,  $f_k$  must be onto as well. Thus,  $f_k$  is a permutation on  $\mathcal{X}$ , and  $D(k, \cdot)$  is the inverse permutation  $f_k^{-1}$ .

**Data Encryption Standard (DES) Algorithm** The Data Encryption Standard (DES) was developed at IBM in response to a solicitation for proposals from the National Bureau of Standards (now the National Institute of Standards). It was published in the Federal Register in 1975 and was adopted as a standard for "unclassified" applications in 1977. The DES algorithm consists of 16 iterations of a simple round cipher. To describe DES it suffices to describe the DES round cipher and the DES key expansion function. We describe each in turn.

(1) The Feistel permutation.

One of the key innovations in DES, invented by Horst Feistel at IBM, builds a permutation from an arbitrary function. Let  $f : \mathcal{X} \rightarrow \mathcal{X}$  be a function. We construct a permutations  $\pi : \mathcal{X}^2 \rightarrow \mathcal{X}^2$  as follows:

$$\pi(x, y) := (y, x \oplus f(y)) \quad (2.12)$$

To show that  $\pi$  is one-to-one we construct its inverse, which is given by:

$$\pi^{-1}(u, v) := (v \oplus f(u), u) \quad (2.13)$$

The function  $\pi$  is called a Feistel permutation and is used to build the DES round cipher. The composition of  $n$  Feistel permutations is called an  $n$ -round Feistel network. Block ciphers designed as a Feistel network are called Feistel ciphers. For DES, the function  $f$  takes 32-bit inputs and the resulting permutation  $\pi$  operates on 64-bit blocks.

(2) The DES round function  $F(k, x)$ .

The DES encryption algorithm is a 16-round Feistel network where each round uses a different function  $f : \mathcal{X} \rightarrow \mathcal{X}$ . In round number  $i$  the function  $f$  is defined as  $f(x) := F(k_i, x)$  where  $k_i$  is a 48-bit key for round number  $i$  and  $F$  is a fixed function called the DES round function. The function  $F$  is the centerpiece of the DES algorithm. And  $F$  uses several auxiliary functions  $E, P$ , and  $S_1, \dots, S_8$ . The function  $E$  expands a 32-bit input to a 48-bit output by rearranging and replicating the input bits. For example,  $E$  maps input bit number 1 to output bits 2 and 48; it maps input bit 2 to output bit number 3, and so on. The function  $P$ , called the mixing permutation, maps a 32-bit input to a 32-bit output by rearranging the bits of the input. For example,  $P$  maps input bit number 1 to output bit number 9; input bit number 2 to output number 15, and so on. At the heart of the DES algorithm are the functions  $S_1, \dots, S_8$  called S-boxes. Each S-box  $S_i$  maps a 6-bit input to a 4-bit output by a lookup table. The DES standard lists these 8 look-up tables, where each table contains 64 entries.

(3) The key expansion function.

The DES key expansion function  $G$  takes as input the 56-bit key  $k$  and outputs 16 keys  $k_1, \dots, k_{16}$ , each 48-bits long. Each key  $k_i$  consists of 48 bits chosen from the 56-bit key, with each  $k_i$  using a different subset of bits from  $k$ .

(4) Initial and final permutations.

The complete DES algorithm consists of 16 iterations of the DES round cipher plus initial and final permutations called IP and FP. These permutations simply rearrange the 64 incoming and outgoing bits. The permutation FP is the inverse of IP. IP and FP have no cryptographic significance and were included for unknown reasons.

**Linear Cryptanalysis** Let  $(E, D)$  be a block cipher where data blocks and keys are bit strings. That is,  $\mathcal{M} = \mathcal{C} = \{0, 1\}^n$  and  $\mathcal{K} = \{0, 1\}^h$ .

For a bit string  $m \in \{0, 1\}^n$  and a set of bit positions  $S \subseteq \{0, \dots, n-1\}$  we use  $m[S]$  to denote the XOR of the bits in positions in  $S$ . That is, if  $S = \{i_1, \dots, i_l\}$  then  $m[S] := m[i_1] \oplus \dots \oplus m[i_l]$ . We say that the block cipher  $(E, D)$  has a linear relation if there exist sets of bit positions  $S_0, S_1 \subseteq \{0, \dots, n-1\}$  and  $S_2 \subseteq \{0, \dots, h-1\}$  such that for all keys  $k \in \mathcal{K}$  and for randomly chosen  $m \in \mathcal{M}$ , we have

$$\Pr[m[S_0] \oplus E(k, m)[S_1] = k[S_2]] \geq \frac{1}{2} + \epsilon \quad (2.14)$$

for some non-negligible  $\epsilon$  called the bias. For an "ideal" cipher the plaintext and ciphertext behave like independent strings so that the relation  $m[S_0] \oplus E(k, m)[S_1] = k[S_2]$  holds with probability exactly  $1/2$ , and therefore  $\epsilon = 0$ . Surprisingly, the DES block cipher has a linear relation with a small, but non-negligible bias.

## 2.3. Other Ciphers

**Quantum Exhaustive Search** Surprisingly, on a quantum computer the same exhaustive search problem can be solved in time proportional to only  $\sqrt{|\mathcal{K}|}$ . For block ciphers like AES-128 this means that exhaustive search will only require about  $\sqrt{2^{128}} = 2^{64}$ . As a result, once quantum computers are built, AES-128 will be considered insecure.

**Grover's Algorithm** Suppose we are given a function  $f : \mathcal{K} \rightarrow \{0, 1\}$  defined as follows

$$f(k) = \begin{cases} 1 & \text{if } k = k_0 \\ 0 & \text{otherwise} \end{cases} \quad (2.15)$$

for some  $k_0 \in \mathcal{K}$ . The goal is to find  $k_0$  given only "black-box" access to  $f$ , namely by only querying  $f$  at different inputs.

**Oracle** Suppose we have some type of cryptographic scheme  $\mathcal{S}$  whose implementation makes use of a block cipher  $\mathcal{E} = (E, D)$  defined over  $(\mathcal{K}, \mathcal{X})$ . Moreover, suppose the scheme  $\mathcal{S}$  evaluates  $E$  at various inputs  $(k', a') \in \mathcal{K} \times \mathcal{X}$ ,



and  $D$  at various inputs  $(k', b') \in \mathcal{K} \times \mathcal{X}$ , but does not look at the internal implementation of  $E$ . In this case, we say that  $\mathcal{S}$  uses  $\mathcal{E}$  as an oracle.

**Chosen Plaintext Attack** A cipher  $\mathcal{E}$  is called semantically secure against chosen plaintext attack, or simply CPA secure, if for all efficient adversaries  $\mathcal{A}$ , the value  $\text{CPAadv}[\mathcal{A}, \mathcal{E}]$  is negligible, where  $\mathcal{A}$ 's advantage with respect to  $\mathcal{E}$  is defined as

$$\text{CPAadv}[\mathcal{A}, \mathcal{E}] := |\Pr[W_0] - \Pr[W_1]| \quad (2.16)$$

**Nonce-based Encryption** Instead of maintaining internal states, both the encryption and decryption algorithms take an additional input  $n'$ , called a nonce. The syntax for nonce-based encryption becomes

$$c = E(k, m, n') \quad (2.17)$$

where  $c \in \mathcal{C}$  is the ciphertext,  $k \in \mathcal{K}$  is the key,  $m \in \mathcal{M}$  is the message, and  $n' \in \mathcal{N}$  is the nonce. Moreover, the encryption algorithm  $E$  is required to be deterministic. Likewise, the decryption syntax becomes

$$m = D(k, m, n') \quad (2.18)$$

The intention is that a message encrypted with a particular nonce should be decrypted with the same nonce — it is up to the application using the encryption scheme to enforce this. We say that such a nonce-based cipher  $\mathcal{E} = (E, D)$  is defined over  $(\mathcal{K}, \mathcal{M}, \mathcal{C}, \mathcal{N})$ .

**Integrity Mechanism Principle** Providing message integrity between two communicating parties requires that the sending party has a secret key unknown to the adversary.

**Message Authentication Codes** A Message Authentication Codes (MAC) system  $\mathcal{I} = (S, V)$  is a pair of efficient algorithms,  $S$  and  $V$ , where  $S$  is called a signing algorithm and  $V$  is called a verification algorithm. Algorithm  $S$  is used to generate tags and algorithm  $V$  is used to verify tags.

(1)  $S$  is a probabilistic algorithm that is invoked as  $t \xleftarrow{R} S(k, m)$ , where  $k$  is a key,  $m$  is a message, and the output  $t$  is called a tag.

(2)  $V$  is a deterministic algorithm that is invoked as  $r \leftarrow V(k, m, t)$ , where  $k$  is a key,  $m$  is a message,  $t$  is a tag, and the output  $r$  uses either accept or reject.

(3) We require that tags generated by  $S$  are always accepted by  $V$ ; that is, the MAC must satisfy the following correctness property: for all keys  $k$  and all

$$\Pr[V(k, m, S(k, m)) = \text{accept}] = 1 \quad (2.19)$$

As usual, we say that keys lie in some finite key space  $\mathcal{K}$ , messages lie in a finite message space  $\mathcal{M}$ , and tags lie in some finite tag space  $\mathcal{T}$ . We say that  $\mathcal{I} = (S, V)$  is defined over  $(\mathcal{K}, \mathcal{M}, \mathcal{T})$ .

**MAC Security** A MAC system  $\mathcal{I}$  is secure if for all efficient adversaries  $\mathcal{A}$ , the value  $\text{MACadv}[\mathcal{A}, \mathcal{I}]$ , denoted as the probability that  $\mathcal{A}$  wins the game, is negligible.

## 3. System Model

### 3.1. Preliminaries and Notations

Throughout this thesis, we use the preliminaries and notations listed as follows.

Table 3.1: Preliminaries and Notations

Notations	Definitions
$\{0, 1\}^n$	The set of all binary strings of length $n$
$\{0, 1\}^*$	The set of all finite binary strings
$x \leftarrow \chi$	An element $x$ being sampled from a distribution $\chi$
$[n]$	The set of integers $\{1, \dots, n\}$
$x \leftarrow X$	An element $x$ being sampled at random from a set $X$
$x \leftarrow \mathcal{A}$	The output $x$ of a probabilistic algorithm $\mathcal{A}$
$x := \mathcal{B}$	The output $x$ of a deterministic algorithm $\mathcal{B}$
$\mathbf{v}$	A sequence of elements
$\mathbf{v}[i], \mathbf{v}_i$	$i^{th}$ element of $\mathbf{v}$
$\#\mathbf{v}$	Total number of elements of $\mathbf{v}$
$\#S$	The cardinality of set $S$
$W$	The universe of words
$f = (w_1, \dots, w_m)$	A file $f$ containing words $w_i$
$\#f$	Total number of words of $f$
$ f $	Bit length of $f$
$\bar{f}$	The file that results from removing all duplicates from $f$
$ s $	Bit length of a string $s$
$\langle s_1, \dots, s_n \rangle$	The concatenation of $n$ strings
$\#\mathbf{A}$	Total number of cells of array $\mathbf{A}$
$\mathbf{A}[i] := v$	Storing $v$ at location $i$ in $\mathbf{A}$
$\#\mathbf{T}$	The number of pairs $(s, v)$ in dictionary (key-value store) $\mathbf{T}$
$\mathbf{T}[s] := v$	Storing the value $v$ under search key $s$ in $\mathbf{T}$
$\#\mathbf{L}$	Total number of nodes
$\mathbf{L}[n] := v$	Storing the value $v$ to node $n$ in $\mathbf{L}$
$\perp$	Empty
$\mathcal{CS}$	Computing server
$\mathcal{ES}$	Encrypting server
$\mathcal{SS}$	Semantic storage server
$\mathcal{VS}$	Visual storage server
$D$	Dataset of all videos in plaintext
$C$	Dataset of all videos in ciphertext
$K'$	Key for encrypting semantic keyword

Continued on next page

Table 3.1 . Continued from previous page

Notations	Definitions
$K^*$	Key for encrypting frame vector
$K_S$	Key for turning $S'$ into $S^*$
$M$	Semantic keywords from videos
$F$	Keyframes from videos
$V$	Frame vectors
$S$	Unique squence connecting keyword and vector
$S'$	Unique encrypted squence in $\mathcal{SS}$
$S^*$	Unique encrypted squence in $\mathcal{VS}$
$I$	Semantic keyword index set
$J$	Frame vector index set
$T'$	Semantic keyword token set
$T^*$	Frame vector token set
$Dis_H$	Haming distance
$Q$	Encrypted videos of interest
$B$	Videos of interest in plaintext form

Note: We follow some definitions in [9].

### 3.2. Overview

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

### 3.3. Functions and Characteristics

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

- $(K', K^*, K_S) \leftarrow Gen(1^\lambda)$ :  $Gen(\cdot)$  With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.



Figure 3.1: System Model

- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

*Correctness* With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

## 4. Construction

### 4.1. Video Preprocess

#### 4.1.1. Framework



Figure 4.1: Video Preprocess

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

(1) With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

(2) Next, the second step is obtaining visual representation for ranking the related videos, which can be summerized below:

i. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically. the part ?? for generalization of our prototype.

ii. With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

### 4.1.2. Algorithms

**Scale-invariant Feature Transform (SIFT)** [10] With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

(1) Scale-space extrema detection.

Search for stable features across multiple scales on basis of a continuous function of scale. The scale space of an image is a function  $L(x, y, \sigma)$  that is produced from the convolution of a Gaussian kernel (at different scales)  $G(x, y, \sigma)$  with the input image  $I(x, y)$ :

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (4.20)$$

where  $*$  is the convolution operation in  $x$  and  $y$ , and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (4.21)$$

and therefore,

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2 \nabla^2 G \quad (4.22)$$

(2) Keypoint localization.

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

$$H_x = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4.23)$$

where  $D(c, y, \sigma)$  is Difference-of-Gaussian. Consider  $(n, g)$  as public parameters while the pair  $(p, q)$  (or  $\lambda$ ) remains private. The crypto-algorithm is depicted in the following.

#### Encryption

plaintext  $m < n$

select  $r < n$

ciphertext  $c = g^m \cdot r^n \bmod n^2$

#### Decryption

ciphertext  $c < n^2$

plaintext  $m = \frac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} \bmod n$

*Correctness* According to equation,

$$\frac{L(w^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} = \frac{\lambda \llbracket w \rrbracket_{1+n}}{\lambda \llbracket g \rrbracket_{1+n}} = \frac{\llbracket w \rrbracket_{1+n}}{\llbracket g \rrbracket_{1+n}} = \llbracket w \rrbracket_g \bmod n \quad (4.24)$$

the correctness can be verified.



## 4.2. First-step Dynamic Keyword Search

### 4.2.1. Framework

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

**Dynamic SSE** With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

- $K \leftarrow \text{Gen}(1^\lambda)$ : With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.
- $(\gamma, \mathbf{a}) \leftarrow \text{aaa}(K, \mathbf{q})$ :  $\mathbf{aa}$ .  $\text{aaaaaa } \gamma$ ,  $\text{aaaaaaaa } \mathbf{a}$ .
- $\tau_s \leftarrow \text{aaaaa}(a, a)$ : is a (possibly probabilistic) algorithm that takes as input a secret key  $a$  and a keyword  $a$ . It outputs a search token  $\tau_s$ .
- $(\tau_a, a) \leftarrow \text{aaaaaa}(a, a)$ :  $\text{aaaaaaaa } a$  and a unique sequence  $a$ .

$\text{aaaaaaaa}$  With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

It is difficult.

### 4.2.2. Algorithms

We use a dynamic SSE scheme  $\text{SSE} = (\text{Gen}, \text{Enc}, \text{SrchToken}, \text{AddToken}, \text{DelToken}, \text{Search}, \text{Add}, \text{Del}, \text{Dec})$   $\text{aaaaaaaa}$ :

**Gen**( $1^\lambda$ ):

$\lambda$ -bit strings  $K_1, K_2, K_3$  is sampled uniformly at random

$K_4 \leftarrow \text{SSE.Gen}(1^\lambda)$

$K = (K_1, K_2, K_3, K_4)$

**SrchToken**( $K, w$ ):

Output  $\tau_s := (F_{K_1}(w), G_{K_2}(w), P_{K_3}(w))$

## 4.3. Second-step Similarity Search

### 4.3.1. Framework

With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

- $K^\circ \leftarrow \text{SlarGen}(1^\lambda)$ : is a probabilistic algorithm that takes as input a security parameter  $\lambda$  and outputs a secret key  $K^\circ$ .

*Correctness-2* With the rapid growth of Internet, various form data is being created and distributed every day. Specifically, the number of videos shared on the Internet, no matter for study, work, or fun, grows dramatically.

### 4.3.2. Algorithms

**AbCd**( $1^\lambda$ ):

$\lambda$ -bit strings  $K_1^\circ, K_2^\circ, K_3^\circ$  is sampled uniformly at random

$K_D^\circ \leftarrow \text{SSE.Gen}(1^\lambda)$

$K^\circ = (K_1^\circ, K_2^\circ, K_3^\circ, K_D^\circ)$

5. Implementation

5.1. Environment

aaaaaaaa

Table 5.1: Environment of Implementation

Name	Type
Processor	Intel(R) Core(TM) i7-4710MQ CPU @ 2.5GHZ
RAM	8.00GB
System	Windows 10 Home 64bit
Logic processors	8
Kernel	4

5.2. Results

aaaaaa.

aaa UCF101 [11] aaaaaaaaaaaaaaaaaaaaaaa <http://crcv.ucf.edu/data/UCF101.php>

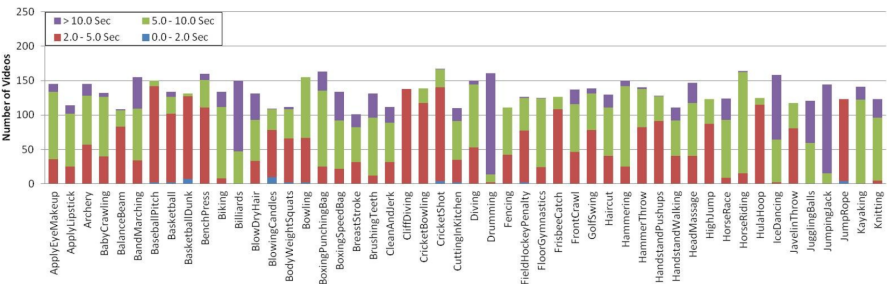


Figure 5.1: UCF101 Videos (1)

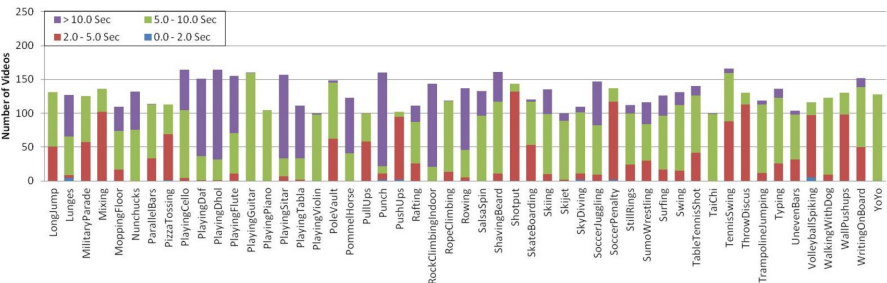


Figure 5.2: UCF101 Videos (2)

aaaaaaaaaaaaaaaaaaaaa Fig.5.3.  
aaaaaaaaaaaaa Fig.5.4, 5.5. aaaaaaaaaaa

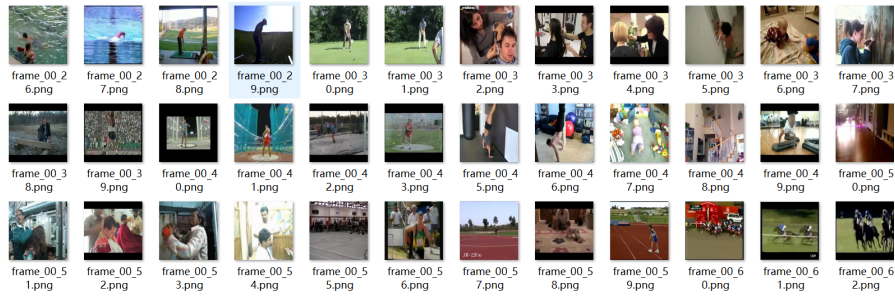


Figure 5.3: Keyframes of UCF101

```

vector383.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
0.000035 0.000190 0.004870 0.069470 0.172003 0.010188 0.000123
0.000057 0.070443 0.037332 0.019736 0.045632 0.171884 0.015363
0.002349 0.011399 0.116053 0.025061 0.004723 0.000092 0.023963
0.026370 0.010879 0.052592 0.000180 0.000063 0.000053 0.000016
0.241988 0.193163 0.003649 0.002196 0.000451 0.001398 0.008296
0.039941 0.107317 0.123436 0.055279 0.021564 0.118273 0.048196
0.007649 0.022126 0.172936 0.236500 0.012035 0.012053 0.241988
0.045422 0.000914 0.000051 0.013328 0.084181 0.081068 0.111329
0.008507 0.000500 0.000000 0.000000 0.202005 0.241988 0.058805
0.014204 0.001966 0.003792 0.000009 0.000019 0.033840 0.101155
0.043911 0.009744 0.049387 0.002667 0.000040 0.000011 0.075598
0.241988 0.057185 0.031344 0.241988 0.082197 0.004486 0.002670
0.023511 0.084250 0.072538 0.116198 0.052017 0.036027 0.009533
0.033012 0.241988 0.215745 0.019746 0.010581 0.004626 0.015754
0.018324 0.010793 0.008794 0.013828 0.000035 0.000038 0.001207
0.000330 0.069011 0.085246 0.025371 0.097700 0.018092 0.003998
0.077139 0.040088 0.219880 0.118949 0.004931 0.026077 0.027151
0.026963 0.018837 0.025164 0.195734 0.241988 0.083388 0.012444
0.000000 0.000293 0.068165 0.052211 0.009952 0.017779 0.062071
0.034016 0.001096 0.008718 0.194653 0.024523 0.000090 0.001518
0.025760 0.019149 0.016106 0.089475 0.003568 0.000000 0.000000
0.013602 0.086206 0.062630 0.021321 0.009920 0.000000 0.000000
0.000000 0.001500 0.258397 0.088951 0.000603 0.000000 0.096821
0.020080 0.000633 0.000387 0.102374 0.195174 0.021391 0.047490
0.258397 0.028477 0.000071 0.000480 0.009648 0.064687 0.079664

```

Figure 5.4: Index of a Video

aaaaaaaaaaaaaaaa

In the Fig.5.6, aaaaaaaaaaaaaaaaaa













 vector380.txt	2018/3/25 18:50	文本文档	1,016 KB
 vector381.txt	2018/3/25 18:50	文本文档	332 KB
 vector382.txt	2018/3/25 18:50	文本文档	393 KB
 vector383.txt	2018/3/25 18:50	文本文档	224 KB
 vector384.txt	2018/3/25 18:50	文本文档	1,000 KB
 vector385.txt	2018/3/25 18:50	文本文档	731 KB
 vector386.txt	2018/3/25 18:50	文本文档	706 KB
 vector387.txt	2018/3/25 18:51	文本文档	604 KB
 vector388.txt	2018/3/25 18:51	文本文档	866 KB
 vector389.txt	2018/3/25 18:51	文本文档	221 KB
 vector390.txt	2018/3/25 18:51	文本文档	599 KB
 vector391.txt	2018/3/25 18:51	文本文档	390 KB

Figure 5.5: Indexes of Videos

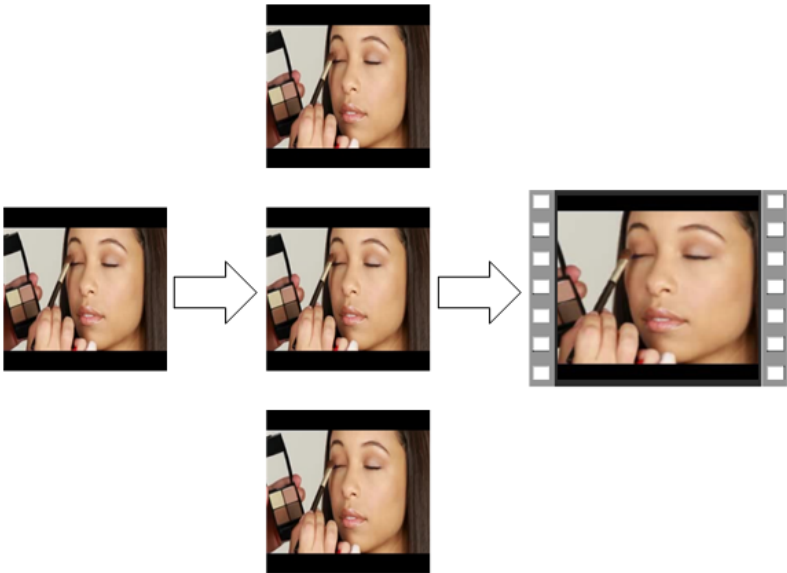


Figure 5.6: Search Results



## 6. Conclusions and Outlook

### 6.1. Conclusions

In my final year project, I have implemented aaaaaaaa

- aaaaaaaa [12],[13],[14] aaaaaaaaaaaaaa
- Secondly, aaaaaaaa
- aaaaa (i) aaaaaaaaa. (ii) aaaaaaaaa
- aaaaaaaa

aaaaaaa

### 6.2. Reflection and Outlook

In my three-month project for bachelor degree, aaaaaaaaaaaa

(1) Read many papers in the field of information retrieval and searchable encryption.

(2) Attend courses (Cryptography 1, Bitcoin and Cryptocurrency Technologies) on coursera.

(3) Attend part of courses (Secure Software Engineering by Sherman S.M. CHOW, Cryptography by Andrej Bogdanov).

(4) Implement a initial and acceptable searchable encryption scheme for videos.

(5) Design a modified first-general-prototype after comparation, which enables searching and ranking related videos.

After my final-year project, I still have much needed works to complete this project finally.

- (1) aaa
- (2) AAAA





# References

- [1] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Trans. Dependable Sec. Comput.*, vol. 13, no. 3, pp. 312–325, 2016.
- [2] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 2, pp. 340–352, 2016.
- [3] Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li, "Multi-probe LSH: efficient indexing for high-dimensional similarity search," in *The 33rd International Conference on Very Large Data Bases*, pp. 950–961, 2007.
- [4] B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud," in *IEEE Conference on Computer Communications, INFOCOM*, pp. 2112–2120, 2014.
- [5] P. N. Aswani and K. C. Shekar, "Fuzzy keyword search over encrypted data using symbol-based trie-traverse search scheme in cloud computing," *CoRR*, vol. abs/1211.3682, 2012.
- [6] Minghui, M. Zhang, Q. Wang, S. S. Chow, M. Du, Y. Chen, and C. Li, "Instantcryptogram: Secure image retrieval service," in *IEEE International Conference on Computer Communications, INFOCOM*.
- [7] D. Boneh and V. Shoup, "A graduate course in applied cryptography," *Version 0.1*, from <http://cryptobook.net>, 2008.
- [8] J. Katz and Y. Lindell, *Introduction to modern cryptography*. CRC press, 2014.
- [9] S. Kamara, C. Papamanthou, and T. Roeder, "Dynamic searchable symmetric encryption," in *the ACM Conference on Computer and Communications Security, CCS*, pp. 965–976, 2012.
- [10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [11] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, vol. abs/1212.0402, 2012.
- [12] Y. Ye, Z. Zhao, Y. Li, L. Chen, J. Xiao, and Y. Zhuang, "Video question answering via attribute-augmented attention network learning," in *The 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 829–832, 2017.
- [13] M. H. T. de Boer, "Semantic mapping in video retrieval," vol. 51, pp. 161–162, 2017.
- [14] Y. Yin, R. Thapliya, and R. Zimmermann, "Encoded semantic tree for automatic user profiling applied to personalized video summarization," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 28, no. 1, pp. 181–192, 2018.



## Acknowledgement

注：本部分大致按时间顺序，无重要性区分。

岁月如梭，韶光易逝，四年的大学时光在不经意间悄悄溜走。致谢中文比较容易写出感情哦

感谢aaaaaaaaaaaaaaaaaaaaa

感谢aaaaaaaaa

感谢aaaaaaaaa

今当远离，不知所言；天涯海角，愿君安好；一路锋芒，一路辉煌。

写于2018年5月20日凌晨

## Achievements

### (1) 论文和专利

i. Chong Fu, **Yu Zheng**, Min Chen, and Zhan-kao Wen. "A color image encryption algorithm using a new 1-D chaotic map", IEEE 17th International Conference on Communication Technology (ICCT), 2017. (Best Paper Award)

ii. 付冲, **郑宇**, 何兴文. 一种具有与明文相关密钥流生成机制的混沌图像加密方法. (发明专利)

### (2) 研究和项目经历

i. 2018年3月 - 2018年5月

Junior research assistant for searchable encryption supervised by Sherman S.M. CHOW.

ii. 2016年8月 - 2017年4月

基于隐写术和混沌密码学的保密通信系统. (项目负责人, 资助8.6万, 大学生创新创业系列之重大项目)

### (3) 竞赛经历

i. 美国大学生数学建模大赛二等奖

ii. 全国大学生数学建模大赛二等奖

iii. 全国大学生信息安全竞赛三等奖

iv. 辽宁省大学生数学建模大赛一等奖

v. 辽宁省大学生电子设计大赛二等奖